

# Development of Software to Automate the Sorting of Spectra

Forrest G. Sedgwick

Advanced Light Source, Ernest Orlando Lawrence Berkeley National Laboratory,  
University of California, Berkeley, California 94720, USA

## INTRODUCTION

The current spectral analysis tools in use at beamline 1.4 emphasize processing of spectral data, rather than analysis of the relationships between groups of spectra. Simple artificial intelligence software has been under development that will assist users who wish to quickly sort large numbers of spectra into groups, without knowing *a priori* what properties are shared by members of those groups. The SpectrumAnalyzer program was designed to automate this task. SpectrumAnalyzer is a modular program, so sorting algorithms with any set of properties can be custom built and easily integrated with the rest of the software. Each algorithm has its own definition of “similar” spectra and can emphasize speed or comprehensiveness as is required. Each algorithm also has its own graphical user interface (GUI), a set of controls and displays unique to that algorithm. SpectrumAnalyzer algorithms feature a user-friendly set of controls and a simple graphical output format so that users may begin work without learning the intricacies of an algorithm’s operation. More advanced users may augment the simple controls and displays with a more detailed set of parameters and calculation results. The spectral data used by the algorithms is imported directly from OMNIC (ThermoNicolet; Madison, WI), the spectral analysis software the users are already familiar with.

The algorithms written emphasized sorting without any assumptions regarding the composition of the spectra to be sorted. The relations between the spectra were unknown, and so the first task of the algorithms was to select spectra which would represent each group. The spectra are then sorted into groups based on some measure of their similarity to the representative spectra.

The first algorithm, the Euclidean Distance Sorter, was very simple and was developed primarily to test the modular design of the program. Most of the development time outside the main program was spent on the second algorithm, the Euclidean Dot Product Sorter [1]. These two modules typify the types of sorting algorithms under development, both in terms of function and user interface.

Figure 1 shows input controls and output displays. In this example, the Euclidean Dot Product Sorter has chosen to sort 144 spectra into 3 groups; two of these groups are very distinct (blue and green) and are composed of the majority of spectra, while the third group is small (only one spectrum) and colored according to its relation to the other two major groups. To the right a color-coded list of the spectra that represent each group allows

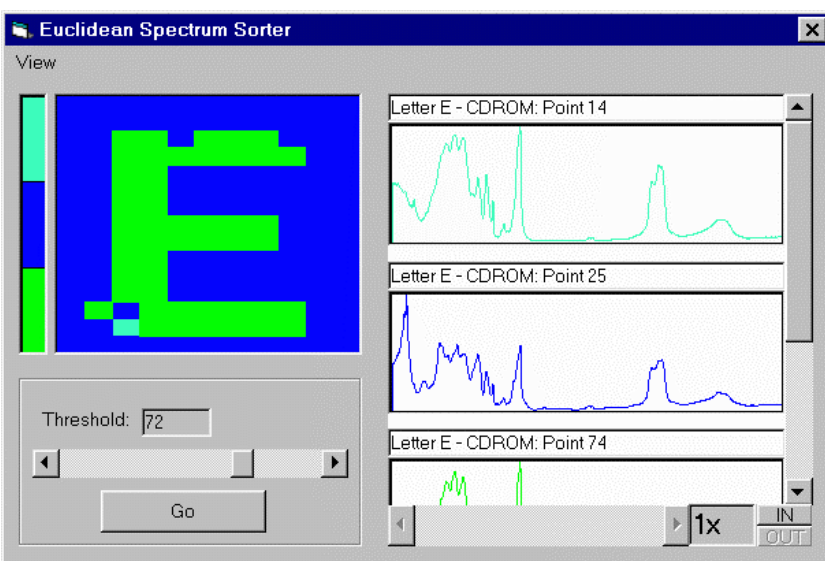


Figure 1. Screen capture of Euclidean Dot Product Sorter showing typical GUI.

the user to see the primary differences between the groups. In this example the spectra in the two main groups differ primarily in the left half of the spectrum. The bottom-left of the GUI contains the most basic controls specific to this algorithm. Above the controls is a false color map, generated whenever the spectra are from points in a spectral map.

## TECHNICAL DETAILS

All the algorithms currently considered rely on representing spectra as points in N-dimensional space, where “N” is the number of wavelengths in each spectrum. The first and simplest sorter developed used the inverse of the Euclidean distance between any two points as a “hit index” representing the degree of similarity between the two spectra. The hit indices of all spectra were stored in an  $n \times n$  matrix, where “n” is the number of spectra. The entries along the diagonal were set to zero (instead of infinity). Selection of the most representative spectrum was the next step, accomplished by adding up all the hit indices in each column of the matrix and selecting the spectrum from the column with the highest total. All spectra with a distance to the representative spectrum that was less than a certain threshold were placed in the same group. As each spectrum was moved into the group, the entries in its corresponding row and column in the hit matrix were zeroed. This removed those spectra already in a group from consideration in the next group selection. The next group selection was made with the same rules as the first. The threshold distance was set manually as a percentage of the maximum distance between any two spectra.

The next sorter, the Euclidean Dot-Product Sorter [1], offered several improvements on the first. First, the vectors representing each spectrum were normalized. The Euclidean distance between points was no longer a good hit index because the resulting points were on the surface of an N-dimensional hyper-sphere. A better measure of similarity was the normalized Euclidean dot-product, the cosine of the “angle” between two points. This hit index varied between 0 (completely different) and 1 (exactly the same). The most representative spectra were chosen in the same way as for the Euclidean Distance Sorter, but the grouping method was different. Spectra were sorted in descending order according to their hit index relative to the most representative spectrum. The hit indices were expected to remain fairly constant at the beginning of the list and then beginning to drop off after a certain point. A cutoff point was selected at the place where the indices were dropping by more than a threshold amount. All spectra with hit indices greater than or equal to the cutoff point were included in the group and had their rows and columns in the hit matrix zeroed. The selection process was then repeated.

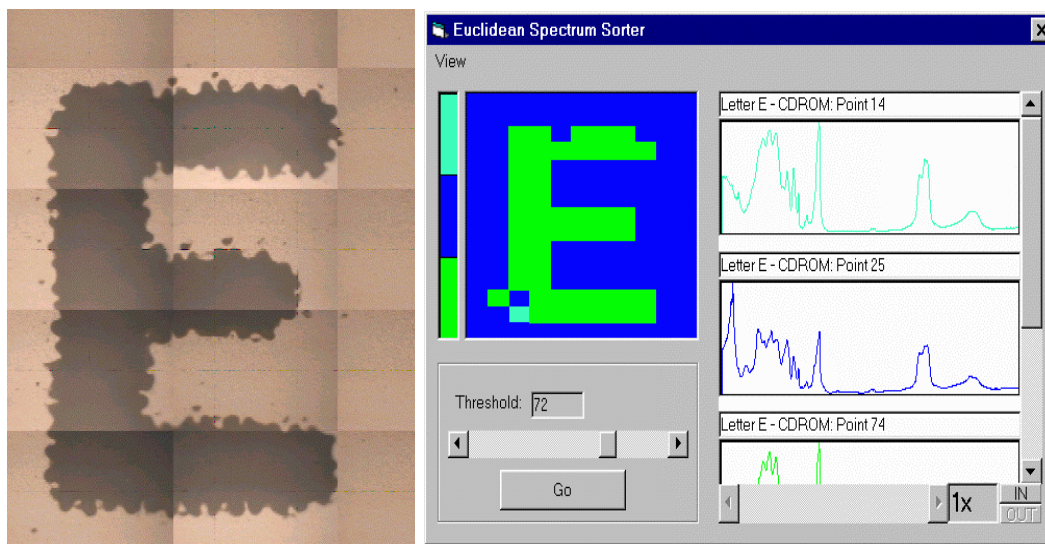


Figure 2. a. Left, photomicrograph of letter “E” burned onto CD-ROM. b. Right, results of processing 144 spectra from the “E” with Euclidean Dot-Product Sorter.

This second sorter performed very well with its test spectra. The results are shown in Figure 2. Figure 2a is a photograph taken through a microscope of a letter "E" burned onto a CD-ROM [2]. A 12x12 grid of spectra was mapped over the "E" and the resulting group of 144 spectra was processed by the Euclidean Dot-Product Sorter. The results, shown in Figure 2b, are very consistent with the photomicrograph in Figure 2a. The sorter has divided the collection into three groups, two main groups colored blue and green and a third group which is colored according to its representative spectrum's distance to the representative spectra of the two main groups.

## FUTURE CONCEPTS

The primary problem with the Euclidean Dot-Product Sorter is that it assumes distributions of points (spectra) in a group to be isotropic in

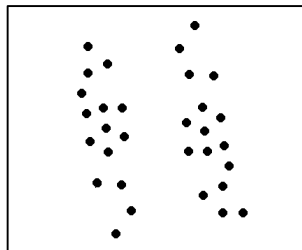


Figure 3a.

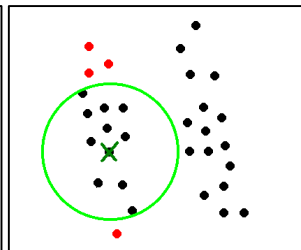


Figure 3b.

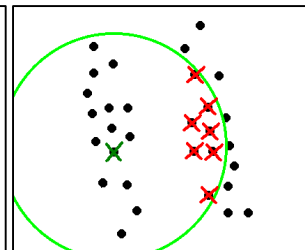


Figure 3c.

N-space. Figure 3 illustrates the problem. In Figure 3a, two groups of points are plotted on a two dimensional slice through N-space. Figure 3b shows a situation where the threshold distance was chosen too small; some points (colored red) are left out of the group. In Figure 3c the threshold distance was chosen large enough that all the necessary points are included in the group, but so are others which shouldn't be (marked with red X's).

The Cluster Sorter would work around this difficulty. The operation of the Cluster Sorter is shown in Figure 4. Beginning with Figure 4a, a representative spectrum is chosen. Spectra within

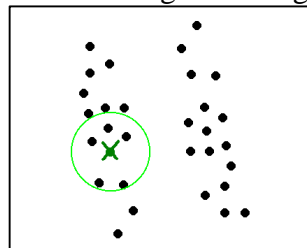


Figure 4a.

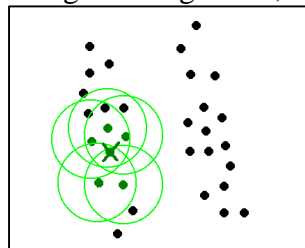


Figure 4b.

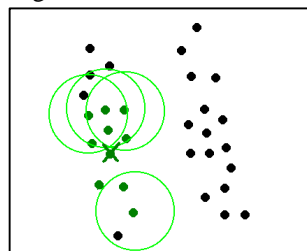


Figure 4c.

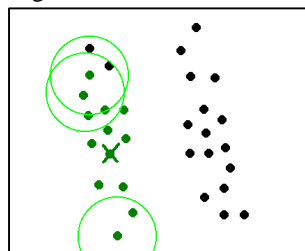


Figure 4d.

a small angular distance of the representative spectrum are grouped together. In the next step, Figure 4b, any spectra within the same small threshold distance from the newly added members of the group are also added to the group. This process continues (Figures 4c and 4d) until no further spectra are added to the group. At that point the grouped spectra are zeroed out of the hit matrix and another representative spectrum is chosen. The Cluster Sorter is shape independent, it takes into account the fact that a group will probably have considerable variation on some wavelengths (dimensions) but will be defined by a narrow range of values for other wavelengths.

## REFERENCES

1. Ideas for this algorithm taken from: M. Diem, L. Chiriboga, A. Pacifico, S. Boydston-White, and H. Yee; in *Biomedical Spectroscopy: Vibrational Spectroscopy and Other Novel Techniques*, Anita Mahadevan-Jansen, Gerwin J. Puppels, Eds., SPIE Vol. 3918 (2000).
2. Photomicrograph and spectra are from sample Atlas map packaged with software.

This work was supported by the Director, Office of Energy Research, Office of Basic Energy Sciences, Materials Science Division, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

Principal investigator: Forrest G. Sedgwick, Advanced Light Source, Ernest Orlando Lawrence Berkeley National Laboratory. Email: sedgwick@cory.EECS.Berkeley.edu Telephone: 510-495-2231.